Brazilian Journal of Experimental Design, Data Analysis and Inferential Statistics

Federal University of Rio de Janeiro

BJEDIS, Rio de Janeiro, Special Edition, v. 01, 2021, Page:69

DOI: https://doi.org/10.29327/232092.1.1-11

ISSN: 2763-6925

Special Edition | Received 02/10/2021. Revised 02/18/2021. Accepted 03/17/2021. Published 05/13/2021.

Data Augmentation Technique in Neural Network Training

Igor Rossi Fermo*[1], Franklin César Flores[2], Cid Marcos Gonçalves Andrade[1]

[1]Chemical Engineering Department, State University of Maringá, Maringá, Brazil; [2]Informatics Department, State University of Maringá, Maringá, Brazil;

Abstract: The data augmentation technique is used to increase the number of images in an image bank for training a neural network. The technique generates new images from an original image, using elementary operations such as rotation, shift, zoom, noise, contrast enlargement and translation. The new images created are different from the original image, even having the same image as the source, the operations used make the image different when compared point by point with the original image, which provides the neural network with a greater number of possibilities for your training. The data augmentation technique is widely used in cases in which the training set is very small and is not sufficient for the neural network to extract the characteristics of a given class. The technique was used to enlarge an image bank of orange photos that will be classified by a Hopfield network with respect to quality and size criteria. Due to the scarcity of images of bad oranges, in order to balance the image bank so that the analysis of results is coherent, the technique was applied in a set with 59 images of oranges, being 50 good and 9 bad. The image bank was expanded to 100 oranges, 50 good and 50 bad. The results obtained were satisfactory and consistent with a high accuracy classifier.

Keywords: Neural network, data augmentation, Image bank, Hopfield network, Orange classification, Training.

Adherence to the BJEDIS' scope: This work is closely related to the scope of BJEDIS as it presents information about machine learning, data analysis, neural network and mathematical modeling.

*Address correspondence to this author at the Department of Chemical Engineering, State University of Maringá, P.O. Box: 87020-900, Maringá, Brazil; Tel/Fax: ++55(44)3011-4752, E-mails: igor_fermo@hotmail.com

1. INTRODUCTION

A neural network is a machine designed to model the way that the brain performs a particular task or function of interest (1). The ANN is able to learn through experiences acquired in a training stage executed on an elaborated database (2). The basic structure of an ANN is the artificial neuron, whose representation can be seen in Figure 1.

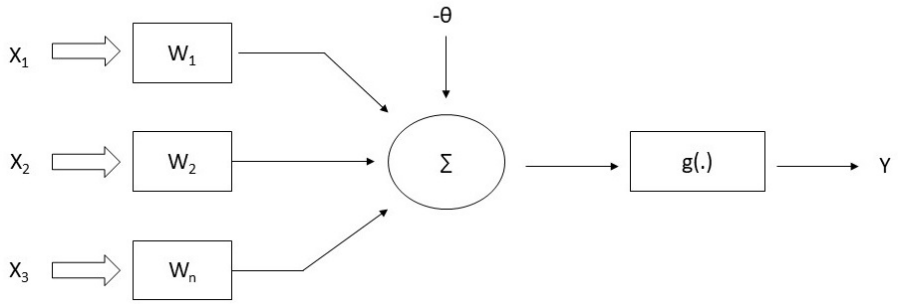

Figure 1 - Basic Structure of an ANN.

The ANN's, to achieve high performance, employ the interconnection of neurons, as in the human brain. Neural networks resemble the human brain in two more aspects, which is the acquisition of knowledge through experiences acquired in a training stage and intensities of interneuron connection, known as synaptic weights, which are used to store the acquired knowledge (1-2).

Initially, in order to perform the neural network training, the synaptic weights must be determined. Synaptic weights are the values used to ponder each of the input variables of the network according to their repeatability in the network database, allowing them to quantify how their relevance in relation to the functionality of the respective neuron (1). During the training stage, input and output variables are related to stabilizing the synaptic weights. Then, after training, it is expected that the ANN will be able to provide an output from a given input.

Hopfield recurrent artificial neural networks have as main characteristic the interconnection of all neurons present in the network that when grouped represents an associative memory for the network support. Due to this feature, this type of network is highly used as associative memories (also named content-addressable memories) that can recover a previously stored pattern from an incomplete or distorted sample (2).

The output of an ideal pattern classifier satisfies two properties. One is the invariance under replacement of a data point with another data point within the same class, we refer to this as the intraclass invariance. The other is the distinction under replacing a data point in one class with a point in another class, and we refer to this as a distinction between classes. Good classifiers have more or less these properties for untrained data (3).

Labeled data is crucial for any supervised machine learning algorithm to work, especially for deep architectures that are easily susceptible to over-tuning (4). Due to the need for training to adjust the parameters of neurons in the neural network, the availability of a robust training set is essential for the satisfactory final performance of the neural network. In addition to the size of the database, an extremely important factor is the balance between classes within the bank, since the network is only able to "learn" after successive presentations of different standards of all the judgments that the network must perform.

In addition to the training stage and as important as it is the neural network validation stage, as it determines whether the neural network really "learned" in a satisfactory way the information that was passed on to it. The validation is also performed with a part of the database, usually with the proportion of 70% for training and 30% for validation, which requires a robust and balanced database, so that the performance of the network can be correctly evaluated.

The problem with small datasets is that the models trained with them do not generalize the validation and test set data well, these problems are known as overfitting (5). In cases where many examples are not available in the database, data augmentation techniques can be used to solve this problem, one of the best known is the data augmentation technique or simply data augmentation. Data augmentation is another way to reduce over-fitting in models, where we increase the amount of training data using information only in our training data (5).

The increase in data allows the generation of a large group of training samples to improve the robustness of the detector (6). The data augmentation field is not new and, in fact, several data augmentation techniques have been applied for specific problems (5). In increasing the data set, the existing data is transformed in some way to create new data that are similar and come from the same (conditional) data generation distribution (7). The data augmentation technique is used to expand the existing data set and can be implemented in a number of ways, such

as linear or nonlinear transformation, addition of auxiliary variable, simulation based on dynamic or evolutionary system and data generation based on model generative (6).

In machine vision, neural networks (NNs) have been used to solve numerous visual recognition problems (8). Regarding the use of ANN for processing fruit images, the techniques normally studied in research works are based on image processing and consider the common parameters for fruit classification (i.e., color, size, format, among others) (9).

In the image processing field, data augmentation originates from traditional data augmentation techniques, such as rotation, translation, shift, zoom, flips, shear, mirror, contrast enlargement and color disturbance (10), which address the issue of limited training data, enriching the training set with original processed samples.

Learning by an increased data set is also beneficial from an engineering point of view. Creating the data set is hard and costly work in product development. The increase in data allows the use of prior knowledge about recognition targets, as it provides easy and cheap substitutes. Second, the quality of virtual data can be easily assessed by human perception. In the case of a visual recognition task, you can check virtual images if they look like a simple visual inspection (3).

These basic ways of increasing data have been widely used in small data sets to combat over-fitting (6). The objective is not only to reduce over-adjustment by means of an increase, but also to increase the data in order to improve the classifier (5). This work aims to demonstrate the results obtained with the application of data augmentation techniques in an unbalanced image bank of oranges whose are classified by a Hopfield recurrent neural network.

2. MATERIALS AND METHOD

The methodology proposed in this work uses the data augmentation technique in an unbalanced database of orange images, composed of large and small oranges in different conservation states, which will later be selected by a recurring Hopfield network with respect to aspects size and quality. This implementation is performed by developing an algorithm using the Matlab® software installed on a Dell Inspiron notebook with an Intel® Core™ i7-4510U CPU 2.6 GHz, and 7.86 Gb of RAM.

2.1. Database Creation

The oranges utilized in the case study are of the *citrus sinensis* (L.) *osbeck* species and belong to the Pêra variety, which is one of the most common at Brazil. Oranges of different sizes were photographed three times in different conservation states to form the image database to be analyzed by RNA. For this, a 13.0-megapixel digital camera was used, capturing the images on a black background. The distance between the camera and the oranges was fixed at 15.3 cm and the fruits were arranged so that they were positioned approximately in the center of the image, for illumination were fixed a LED tape positioned upper the oranges from a distance of approximately 30 cm.

At the end of the process, 59 images of oranges of different sizes and conservation status were obtained. The images are 3120x4160 pixels in size.

The database was divided into two, with one bank classified by size and the other by quality, each with 59 oranges. The bank classified with the size criterion has 39 small oranges and 20 large oranges. The bank classified with the quality criterion has 52 good and 7 bad oranges. The figure 2 shows the 59 oranges that make up the image bank with their respective binary images.

Binary images were used to determine the size of the oranges. The binary images were submitted to a pixel counting algorithm, and through this it was possible to determine which oranges were large and small. As a classification criterion, the average was used, that is, oranges with the number of white pixels above the average were considered large oranges and oranges with the number of pixels below the average were considered small. As can be seen, some oranges have white pixels inside, to avoid counting errors the white pixels inside the image have been removed by means of the morphological operation closing.

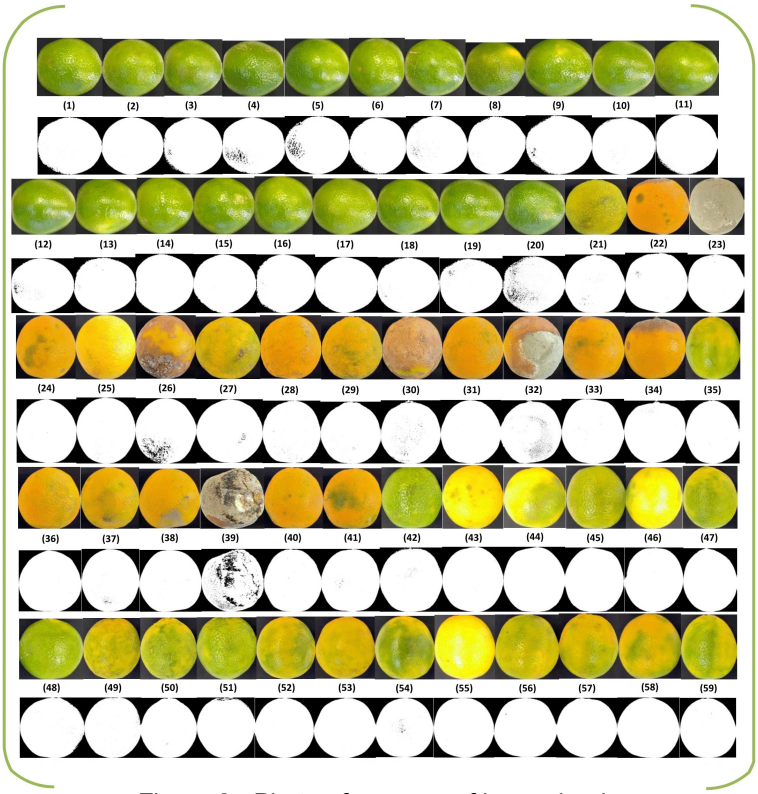

Figure 2 - Photo of oranges of image bank.

2.2. Data Augmentation Technique

After the creation of the image bank, to solve the problem of imbalance between classes, a data augmentation algorithm was used. The algorithm is composed of similar operations such as rotation, translation, horizontal and vertical flip and noise, all determined randomly by the algorithm with the purpose of preventing the same operation from being performed more than once and producing repeated results.

To prevent the algorithm from producing repeated images and generating false results, a point-to-point image comparison algorithm was used to exclude possible identical images

The two banks obtained after the data augmentation technique have 100 oranges each, with 50 small and 50 large oranges in one bank and 50 good oranges and 50 spoiled oranges in another image bank.

2.3 Pre-treatment and ANN working

With the images obtained, two oranges were selected to be the patterns presented to ANN. One image represents the pattern of a good orange, the other image represents a spoiled pattern, or, one representing a large pattern and another representing a small pattern. After that, a third image is received and will be analyzed by ANN.

The pre-treatment of images consists of converting the image obtained in the RGB system's color scale into an 8-bit grayscale image, and crop them to a uniform size in a region de interest and eliminate extra background, what increases the algorithm performance.

After converting to gray tones, the image goes through the binarization process, which consists of replacing each pixel with a value higher than a segmentation threshold with a value of 1 (white) and below that level with a value of 0 (black). The grayscale image was binarized with the help of the Matlab® imbinarize function, in which different parameters were used for the analysis of the size and quality classes.

After binarization, the image will be resized to a matrix with dimensions of 40x40 pixels, where, in the new image, each position of the matrix corresponds to a pixel.

The next step is the pixel-by-pixel comparison of the vectors. The vectors are analyzed by the network and, if it is possible to approximate the image submitted to one of the patterns presented, the ANN returns a positive value. If this approach is not possible, the network returns a negative response.

Finally, the program displays a message indicating the likelihood of the fruit assimilating to the network's standards. At this point, the software displays the number 1 if the fruit resembles pattern number 2 (good or large), or 2 if the fruit resembles pattern number 1 (rotten or small).

A great advantage of this method is the possibility of changing the analysis made by the network, since it is possible that different patterns of good and bad oranges are presented for the network. This possibility becomes an advantage, since it is possible, for example, that a fruit fails in one season, but is approved in another due to factors such as climate, region of production, market requirements, among others. In short, if the quality standard changes, this change can be easily implemented on the network. The algorithm analyzes the visual proximity between two images, in this way, two similar images represent two fruits of the same quality.

Such analyzes can be extended to many aspects, being a function of image processing only. With five different entries, five different patterns can be defined and the user is the one who determines which of these patterns are good or bad. In addition, this approach allows the development of a separation process in many categories of fruit, a fact that expands the possibility of applying the proposed methodology.

Finally, this approach allows for constant changes in quality standards, because if some fruits are classified in the wrong way this can be corrected in the next period of time according to changes that may occur in the various factors that influence the quality of the fruits. Consequently, the algorithm is flexible due to the possibility of changing the concept of a good fruit at any time.

3. RESULTS AND DISCUSSION

The network presented an efficient behavior when comparing the two models. Using the parameters s = 1 and Foreground polarity = bright in the imbinarize function of Matlab®, the results obtained were 85% correct for the quality of the fruit. Figures 3 and 4 show the standards used as a comparison criterion by the ANN for the quality criterion before and after the image binarization process.

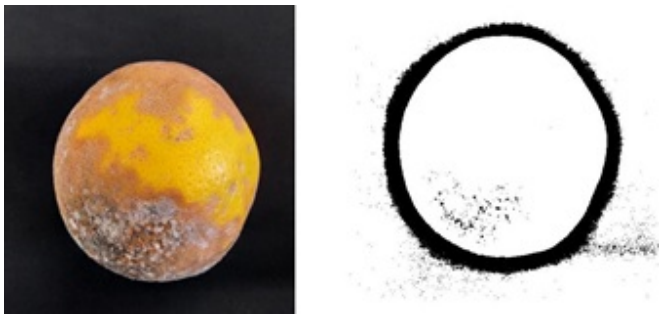

Figure 3 - Pattern of rotten orange.

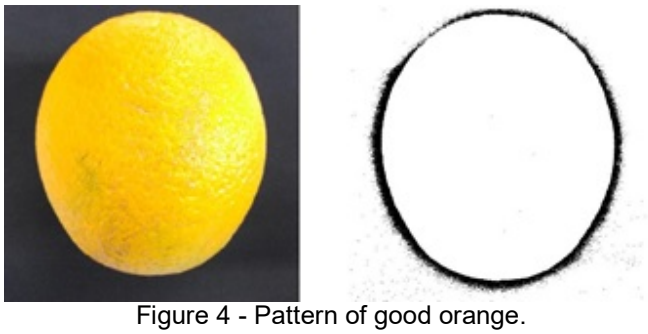

Figure 4 - Pattern of good orange.

Table 1 presents the confusion matrix of the results presented by Hopfield's ANN. For the analysis of the classifier's performance, the two oranges presented as standards for the ANN were excluded from the analysis.

Table **1** - Confusion matrix of classifier with respect to quality

		Prevision		
	Class	Good	Rotten	
Real	Good	37	12	49
	Rotten	3	46	49
		40	58	98

As can be seen in the confusion matrix, the classifier presented satisfactory results regarding the classification of oranges in relation to the quality aspect, as it presents higher values on the main diagonal and reduced values on the secondary diagonal.

When analyzing the size criterion, two images were again provided for the network, one of an orange considered large and another of an orange considered small, the parameters s = 0.3 and Foreground polarity = dark were used in the Matlab® imbinarize function, Figures 5 and 6 shows the oranges used as a large and small pattern, together with their binary image generated by the Matlab® imbinarize function, respectively.

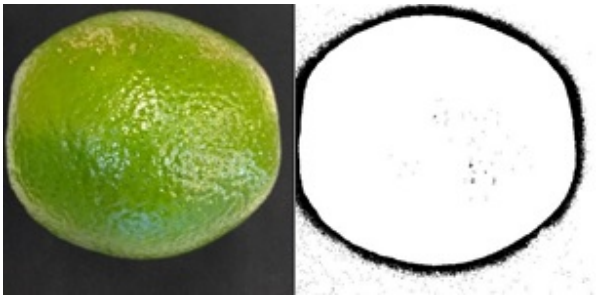

Figure 5 - Pattern of large orange.

Figure 6 - Pattern of small orange.

As can be seen, although the orange considered small does not present a good appearance, the ANN presented satisfactory results, with a percentage of correct answers of 85%, like the quality criteria. Table 2 shows the confusion matrix of the orange classifier in relation to size.

Table **2**. Confusion matrix of classifier with respect to size.

		Prevision		
	Class	Large	Small	
Real	Large	37	12	49
	Small	6	43	49
		43	55	98

As can be seen in Table 2, the confusion matrix presents high values in the main diagonal and low values in the other positions, which indicates a classifier close to the ideal. Table 3 shows the performance measures of the Hopfield ANN classifiers with respect to quality and size class.

Table **3** - Performance measure of classifiers.

Medida de desempenho		
Acuracy	0,85	0,85
Major error	0,5	0,5
Error rate	0,15	0,18
Precision (+)	0,92	0,86
Precision (-)	0,79	0,78
Recall (+)	0,75	0,8
Recall (-)	0,94	0,88
F-Measure (+)	0,83	0,8
F-Measure (-)	0,86	0,83

Regarding both the size of the oranges and the quality, it is clear that the system had a high rate of accuracy on the tested images, reaching an accuracy of 85% for both. The majority error that accuses the balance between classes, in both cases, is 0.5, which indicates that both classes in each criterion have the same number of fruits.

Analyzing the data shown in Table 3, a high accuracy value (85%) is observed and the error rate of the algorithm is below 20%, reinforcing the quality of the Hopfield classifier results. When analyzing the F-Measure values, it appears that the classifier has quality results for both classes, with results above 80%.

5. CONCLUSION

This work presented a data augmentation method applied in a bank initially composed of 59 oranges of different sizes and conservation status. An image capture procedure with 59 oranges was developed so that using Matlab® it was possible to implement a recurrent Hopfield network.

The results obtained showed a good percentage of correct answers (average of 85%), which means that Hopfield's recurrent RNA presents satisfactory results when compared to other existing classifiers.

Finally, the results reinforce the validation of the method used, fulfilling so that the objectives of the study were achieved. For further work, the authors suggest tests of more parameters for classification of fruits in different categories, such as level of rot, number of grooves, among others, which would allow a better and more refined classification of oranges.

LIST OF ABBREVIATIONS

ANN – Artificial neural network

CNN – Convolutional neural network

CONFLICT OF INTEREST

This work was carried out with the support of the Coordination for the Improvement of Higher Education Personnel - Brazil (CAPES) - Financing Code 001.

ACKNOWLEDGEMENTS

We also thank the Department of Chemical Engineering at the State University of Maringá for the enormous intellectual contribution to the development of the work.

REFERENCES

1. Haykin, S., 2009. Neural Networks and Learning Machines, 3rd ed. Pearson, New Jersey, BR.

2. Silva, I.N., Spatti, D.H., Flauzino, R.A., Liboni, L.H.B., Alves, S.F.D., 2017. Artificial Neural Networks – a Practical Course. Springer, Switzerland.

3. Sato, I., Nishimura, H., Yokoi, K., APAC: Augmentted Pattern Classification with Neural Networks. arXiv preprint arXiv:1505.03229, 2015.

4. DeVries, T., & Taylor, G. W. Dataset Augmentation in Feature Space. arXiv preprint arXiv:1702.05538, 2017.

5. Perez L. and Wang J., "The effectiveness of data augmentation in image classification using deep learning," arXiv preprint arXiv:1712.04621, 2017.

6. Han, D., Liu, Q., Fan, W., A new image classification method using CNN transfer learning and web data augmentation. Expert Systems with Applications C 2018.

7. Bengio, Y., Mesnil, G., Dauphin, Y., Rifai, S. Better mixing via deep representations. In ICML (1), pp. 552–560, 2013.

8. Fan, S., Li, J., Zhang, Y., Tian, X., Wang, Q., He, X., Zhang, C., Huang, W., 2020. On line detection of defective apples using computer vision system combined with deep learning methods. J. Food Eng. 286, 110102, http://dx.doi.org/10.1016/j.jfoodeng.2020.110102

9. Fermo, I. R., Cavali, T. S., Bonfin-Rocha, L., Srutkoske, C., Flores, F. C., Andrade, C. M. G., 2021. Development of a low-coast digital image processing system for oranges selection using Hopfield networks. Food and bioproducts processing. 125, 181-192, https://doi.org/10.1016/j.fbp.2020.11.012.

10. Flusser, J. & Suk, T. Pattern recognition by affine moment invariants. Pattern Recognition, 1993 26 (1), 167–174.